IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

ATTORNEY DOCKET NO. **AUS919970761US2**

In re Application of:	§	
	§	Examiner: **Jude Jean Gilles**
JAMES W. ARENDT, ET AL.	§	
	§	
Serial No. **10/042,412**	§	Confirmation No. **9333**
	§	
Filed: **January 7, 2002**	§	
	§	Art Unit: **2143**
For: **HIGHLY SCALABLE AND HIGHLY AVAILABLE CLUSTER SYSTEM MANAGEMENT SCHEME**	§	

APPEAL BRIEF

Mail Stop Appeal Briefs - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This Appeal Brief is submitted in support of an Appeal of the Examiner's final rejection in the above-identified application. A Notice of Appeal was filed in this case on October 29, 2007. Please charge the fee of $510.00 due under 37 C.F.R. § 1.17(c) for filing the brief, as well as any additional required fees, to IBM Deposit Account No. 09-0447.

REAL PARTY IN INTEREST

The real party in interest in the present Appeal is International Business Machines Corporation, the Assignee of the present application as evidenced by the Assignment recorded at reel 009312 and frame 0357.

RELATED APPEALS AND INTERFERENCES

There are no other appeals or interferences known to Appellants, the Appellants' legal representative, or assignee, which directly affect or would be directly affected by or have a bearing on the Board's decision in the pending appeal.

STATUS OF CLAIMS

Claims 1-13, 17, 21-22, 25, 26 and 30 have been cancelled. Claims 14-16, 18-20, 23, 24, 27-29 and 31 stand finally rejected by the Examiner as noted in the Final Office Action dated September 11, 2007.

STATUS OF AMENDMENTS

No amendment has been submitted subsequent to the final rejection that leads to this appeal.

SUMMARY OF THE CLAIMED SUBJECT MATTER

The present invention is directed to a method, computer program product and a cluster multiprocessing system which may be utilized to manage large scale cluster systems. As described in the present specification at page 3, line 12 et seq., in a centralized cluster configuration database the database may be replicated and maintained on a number of data processing systems within the cluster. Thus, in a small cluster, the system configuration and status information may be readily replicated to all data processing systems in the cluster for use by each data processing system in performing system management functions, such as failure recovery and load balancing. Full replication provides a highly available cluster configuration database and performs adequately as long as the cluster size remains small (two to eight data processing systems). In a very large cluster, the cost associated with full replication are prohibitively high. Thus, as described in the present specification at page 5, line 13 et seq., the present invention is directed to a method, computer program product and system for maintaining a distributed database containing cluster configuration information without incurring the costs associated with full replication.

The method set forth within claim 14 for partially replicating configuration information comprises the defining of a subset of data processing systems within a cluster system as a resource group when a data processing system may be a member of more than one resource group. This portion of claim 1 is described in the present specification at page 12, line 14 et seq., with reference to Figure 2A and is illustrated within that Figure. As illustrated, Figure 2A depicts nine data processing systems 202-218 which are organized as four resource groups 220-

226. Thus, one resource group comprises systems 202, 208 and 214, the second resource group comprises systems 204, 210 and 216, the third resource group comprises systems 206, 212 and 218 and the fourth resource group comprises systems 208, 210 and 212. Thus, it may be seen that subsets of the total number of data processing systems are defined as resource groups and that indeed, any particular data processing system may be a member of more than one resource group.

Next, in accordance with an important feature of the present invention, the configuration data for an entire resource group is defined by "instantiating a configuration object containing configuration and status information for a highly available application corresponding to the resource group and having an associated list of data processing systems within the resource group" as depicted in Figure 3 of the present application and as described in the present specification at page 23, et seq. As described therein, upon the occurrence of a configuration/status data change, as illustrated at step 302 of Figure 3, a determination is first made as to whether or not that change is a "cluster-level change." If the change is a cluster-level change, that change is replicated throughout the entire cluster as depicted at step 306. However, in the event the change is not a cluster-level change, the process passes to step 308. Step 308 illustrates the replication of that change among the resource group nodes. Next, the process described within Figure 3 passes to block 310 which illustrates a determination of whether or not a node within the resource group is shared by another resource group. If not, the process passes to block 314 and terminates. If however, a node within the resource is shared by other resource groups, the change is replicated to the other resource group which shares that node as depicted at block 312. Thus, the process described above with respect to Figure 3 illustrates the replication of configuration data only on each data processing system within the resource group.

Next, with reference to claim 23, a computer program product is set forth which comprises a computer program within a computer useable medium which contains instructions for carrying out each step of the method described above with respect to claim 14. Thus, the definition of a subset of data processing systems within a network as a resource group is again illustrated with respect to Figure 2A as described above. Further, the instantiation of a configuration object containing configuration and status information for a highly available application corresponding to that resource group is a defined and replication of configuration data within that resource group are also illustrated within the present specification as described with respect to Figure 3 above. In this manner, the computer program product of claim 23 is described in substantially identical manner within the specification as the method steps of claim 14.

Finally, the cluster multiprocessing system of claim 27 is similarly described in the present specification. A plurality of data processing systems which are segregated into a plurality of resource groups wherein each of the plurality of data processing systems may be a member of more than one resource group is clearly illustrated within Figure 2A and is described in the present specification at page 12, line 14 et seq. As noted above, Figure 2A depicts nine data processing systems 202-218 which are organized as four resource groups 220-226. The configuration objects set forth within claim 27, as those having ordinary skill in the art will appreciate, comprise software elements which are illustrated within Figure 3 of the specification and described at page 23, et seq. Similarly, the replication of each of a plurality of configuration objects only within each data processing system within the resource group is also illustrated within Figure 3 as described above.

GROUNDS TO REJECT AND TO BE REVIEWED UPON APPEAL

The Examiner's rejection of claims 4-16, 18-20, 23, 24, 27-29 and 31 under 35 U.S.C.§102(e) as being anticipated by *Vert et al.* United States Patent Number 6,360,331 is to be reviewed on appeal.

ARGUMENT

The claims of the present application are directed to a method, computer program product and system for enhancing the efficiency of operation of a distributed cluster data processing system. This is accomplished, as generally set forth within each of the independent claims, by defining "resource groups" within a subset of data processing systems within a cluster system and thereafter replicating configuration data for a particular highly available application only on each data processing system contained within the resource group of the cluster system.

Vert et al., the only reference relied upon by the Examiner, is directed to a system for maintaining configuration data in order to provide a redundant computer system by "tracking changes to an application configuration information, detecting those changes and taking a 'snapshot' of the data." That data is thereafter "written to a storage device shared by the systems of the cluster, such as a quorum disk." (See the *Vert et al.* abstract). Thereafter, if the system running that particular application fails, the application is turned over to a second system which must necessarily retrieve status and configuration information from the aforementioned quorum disk in order to reestablish the particular application. This is described in *Vert et al.* at column 2, line 35 et seq., where *Vert et al.* notes "When the application is failed over to a second system, the snapshot for the application is retrieved from the quorum disk by a restore mechanism and written to the registry of the second system in a corresponding subtree. The application is then

run on the second system using the restored application configuration information for that application."

Thus, *Vert et al.* avoids the necessity of replicating data throughout an entire cluster data processing system; however, *Vert et al.* suffers from several noticeable shortfalls. Firstly, a failure of the quorum disk will render the application inoperative should the processing system running that application fail. Secondly, in the event of a failure of the system running that application, the status and configuration for the application must be retrieved by the "restore mechanism" (see above) and then written to the registry of a second system within the cluster before the application can be restored.

In contrast, as expressly set forth within the claims of the present application, by predefining a "resource group" within a cluster system which is responsible for a particular highly available application and thereafter replicating the configuration data for that application only on data processing systems within that resource group, in the event of a failure of the system controlling the highly available application, a second data processing system within the resource group may quickly and efficiently begin operation of that highly available application in a manner not permitted by the system disclosed within *Vert et al.* This is also accomplished, as described in the present specification, in a manner which is much more efficient and cost effective than the replication of configuration data for the application on every system within the cluster.

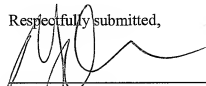
In dismissing Applicant's Request for Reconsideration filed on July 18, 2007 as "not deemed fully persuasive" the Examiner accuses the Applicant of mischaracterizing the teachings of *Vert et al.* and of having "sherry-picked" portions of *Vert et al.* which teach away from the invention without taking into account the essence of the teachings of Vert et al. The Examiner points out quite correctly that *Vert et al.* teach a cluster system; however, nothing within *Vert et al.* anticipates, shows or suggests in any way the designation of a subset of the data processing systems within the cluster as a "resource group" as that term is utilized within the present application.

Indeed, *Vert et al.* implicitly recognizes the difficulty in replicating configuration data for a large cluster within each member of that cluster by teaching the utilization of a quorum disk which may be accessed by any member of the cluster. This is, in the opinion of the Applicant, not anticipatory of, or suggestive of the system set forth within the present claims wherein subsets of the cluster are designated as resource groups and wherein configuration and status information for a highly available application are replicated only within the resource group subset of the cluster.

CONCLUSION

In summary, claims 14, 23 and 27 each set forth expressly and clearly that configuration and status information for a highly available application is replicated on each data processing system within a resource group where the resource group is defined as a "subset of data processing systems within a cluster system..." It is beyond cavil that *Vert et al.* can not be said to anticipate this unclaimed invention as *Vert et al.* specifically and expressly teaches that configuration and status information is not replicated within multiple data processing systems but is in fact replicated to a quorum disk which may be accessed by other data processing systems after a failure has occurred. In view of the fact that the claims of the present application expressly set forth replication within multiple data processing systems within a resource group and wherein that resource group is defined as a subset of data processing systems within a cluster, Applicant respectfully urges that the replication of configuration and status information at a single location within a "quorum disk" as taught by *Vert et al.* can not provide an appropriate basis for a rejection of the present claims and reversal of the Examiner is respectfully requested.

Respectfully submitted,



Andrew J. Dillon
Reg. No. 29,634
DILLON & YUDELL LLP
8911 N. Capital of Texas Hwy., Suite 2110
Austin, Texas 78759
(512) 343-6116
ATTORNEY FOR APPELLANTS

CLAIMS APPENDIX

1-13. (Cancelled)

14. A method of partially replicating configuration information in a distributed database, comprising:

defining a subset of data processing systems within a cluster system as a resource group, wherein a data processing system may be a member of more than one resource group;

defining configuration data for the resource group by instantiating a configuration object containing configuration and status information for a highly available application corresponding to the resource group and having an associated list of data processing systems within the resource group; and

replicating the configuration data only on each data processing system within the resource group.

15. The method of claim 14, wherein the step of defining a subset of data processing systems within a cluster as a resource group further comprises:

defining a highly available application and each data processing system designated to manage the application as a resource group.

16. The method of claim 15, wherein the step of defining a highly available application and each data processing system managing the application as a resource group further comprises:

defining a plurality of resource groups for each highly available application within the cluster, each resource group including all data processing systems managing the corresponding application.

17. (Cancelled)

18. The method of claim 14, wherein the step of replicating the configuration data only on each data processing system within the resource group further comprises:

replicating the configuration object on each data processing system identified in an owners list associated with the configuration object.

19. The method of claim 14, wherein the step of replicating the configuration data only on each data processing system within the resource group further comprises:

replicating, on a data processing system, a configuration object for each resource group including the data processing system.

20. The method of claim 14, further comprising:

maintaining, on a data processing system, a configuration object for each resource group including the data processing system and no configuration objects for other resource groups.

21-22. (Cancelled)

23. A computer program product in a computer usable medium, comprising:

instructions defining a subset of data processing systems within a network as a resource group, wherein a data processing system may be a member of more than one resource group;

instructions defining configuration data for the resource group comprising instructions instantiating a configuration object containing configuration and status information for a highly available application corresponding to the resource group and having an associated list of data processing systems within the resource group; and

instructions for replicating the configuration data only on each data processing system within the resource group.

24. The computer program product of claim 23, wherein the instructions defining a highly available application and each data processing system managing the application as a resource group further comprise:

instructions defining a plurality of resource groups for each highly available application within the network, each resource group including all data processing systems managing the corresponding application.

25-26. (Cancelled)

27. A cluster multiprocessing system, comprising:

a plurality of data processing systems segregated into a plurality of resource groups, wherein each of the plurality of data processing systems may be a member of more than one resource group;

a plurality of configuration objects each corresponding to a resource group within the plurality of resource groups wherein each of the plurality of configuration objects contains configuration and status information for a highly available application corresponding to the resource group and an associated owners list of data processing systems within the resource group; and

wherein each of the plurality of configuration objects is replicated only on each data processing system within the resource group associated with the configuration object.

28. The cluster multiprocessor system of Claim 27, wherein a highly available application and each data processing system designated to manage the application is defined as a resource group.

29. The cluster multiprocessor system of Claim 28, wherein a plurality of resource groups is defined for each highly available application within the cluster, each resource group including all data processing systems managing the corresponding application.

30. (Cancelled)

31. The cluster multiprocessor system of Claim 27, wherein each of the plurality of configuration objects are replicated on each data processing system identified in an owners list associated with the configuration object.

EVIDENCE APPENDIX

None

RELATED PROCEEDINGS APPENDIX

None